

CLOUD-NATIVE MICROSERVICES ARCHITECTURE FOR SECURE AND SCALABLE MORTGAGE APPLICATION

CAPABILITIES SHOWN



Advisory
Services



Cloud
Migration



Digital
Transformation



Cloud
Engineering



Observability
& Monitoring

ABOUT THE COMPANY

We are an ISO 9001:2008, 9001:27001, 20000-1:2018, CMMI Level 3, EDWOSB providing superior, affordable and innovative business management and information technology services to federal and private sector clients nationwide. We specialize in Software Development, Business Intelligence (BI), Data Management, Data Governance, Cyber Security, Data Quality, Master Data Management, Advanced Data Analytics and Cloud Services.



ABOUT THE CUSTOMER – SECONDARY MORTGAGE COMPANY



The Financial Company is a leading source of mortgage financing in all markets and at all times. They ensure the availability of affordable mortgage loans. The financing solutions They develop make sustainable homeownership and workforce rental housing a reality for millions of people.

CUSTOMER MIGRATION TO A CLOUD-NATIVE MICROSERVICES ARCHITECTURE FOR A SECURE AND SCALABLE APPLICATION

Loan Delivery is the mortgage provider's official web-based application that provides the ability to submit loans for whole loan sale, mortgage-backed security (MBS) pools, and As Soon as Pooled Plus (ASAP Plus) loans to this financial company.

Key Challenge

- ✓ Reduce human error risk while performing deployments and speed up releases
- ✓ Need infrastructure to scale in response to surging traffic in response to market conditions.
- ✓ Legacy tech stack with security vulnerabilities and scalability issues

SOLUTION



The application was rearchitected to follow cloud native principles using a microservices architecture on an ECS cluster. All incoming web traffic flows through the iDMZ and API gateway tier. Users are federated and authenticated using an inhouse security product that's wrapped around active directory. All communication between the various components happens over TLS. The file upload is performed by scanning files imported by customer to S3 buckets.

SERVICES USED



- ✓ **Compute and Application Services:**
 - ✓ Amazon ECS (Elastic Container Service): Used for deploying containerized workloads in a microservices architecture.
 - ✓ Amazon Lambda: Used for serverless workloads, event-driven tasks, and batch processing.
 - ✓ AWS Batch: Utilized for processing batch jobs.
 - ✓ Amazon EC2 (Elastic Compute Cloud): Used for virtual machine instances.
 - ✓ AWS Fargate: Employed for containerized workloads without needing to manage the underlying infrastructure.
 - ✓ Amazon API Gateway: Used for creating and managing APIs.



✔ **Data Services:**

- ✔ Amazon RDS (Relational Database Service): Used for managed relational databases, including PostgreSQL and Oracle.
- ✔ Amazon S3 (Simple Storage Service): Employed for object storage and backup.
- ✔ Amazon EFS (Elastic File System): Used for scalable and managed file storage.
- ✔ Amazon DynamoDB: Utilized for NoSQL database needs.
- ✔ Amazon ElastiCache: Used for caching and in-memory data storage.

✔ **Networking and Security Services:**

- ✔ Amazon VPC (Virtual Private Cloud): Utilized for networking isolation and control.
- ✔ AWS Direct Connect: Employed for establishing dedicated network connections between on-premises and AWS.
- ✔ Amazon CloudFront: Used for content delivery and edge caching.
- ✔ Amazon Route 53: Employed for domain name system (DNS) management.
- ✔ AWS WAF (Web Application Firewall): Utilized for protecting web applications from common exploits and vulnerabilities.
- ✔ AWS Shield: Used for protecting against DDoS attacks.
- ✔ AWS Inspector: Employed for assessing application vulnerability and security issues.
- ✔ AWS GuardDuty: Used for intelligent threat detection.

✔ **Monitoring and Logging Services:**

- ✔ Amazon CloudWatch: Used for monitoring AWS resources and collecting logs.
- ✔ AWS X-Ray: Utilized for tracing and analyzing application performance.
- ✔ Splunk: Employed for log analysis and insights monitoring.
- ✔ Dynatrace: Used for application health and performance monitoring.
- ✔ Catchpoint: Employed for user experience monitoring and performance testing.

✔ **Identity and Access Management:**

- ✔ AWS IAM (Identity and Access Management): Used for managing user identities and permissions.

✔ **Deployment and Automation Services:**

- ✔ AWS CloudFormation: Utilized for Infrastructure as Code (IaC) to define and provision resources.
- ✔ Jenkins: Used for continuous integration and continuous delivery (CI/CD) pipelines.
- ✔ Ansible: Employed for configuration management and automation.

✔ **Other Services:**

- ✔ AWS Secrets Manager: Used for storing and managing sensitive information securely.
- ✔ AWS Step Functions: Employed for serverless workflow orchestration.
- ✔ AWS Transit Gateway: Used for connecting multiple VPCs and on-premises networks.

THE BENEFITS

- ✔ Operational Efficiency through Cloud Native Architecture:
 - ✔ By using microservices architecture on an ECS cluster, the application experienced a 25% reduction in response times due to optimized service communication.
- ✔ Enhanced Security Posture:
 - ✔ By migrating to AWS, the application achieved a 30% reduction in security incidents, as measured by the number of security alerts triggered by AWS Inspector and GuardDuty.
- ✔ Improved Development and Deployment Workflow:
 - ✔ The adoption of Infrastructure as Code (IaC) with AWS CloudFormation and CI/CD pipelines resulted in a 40% reduction in deployment time for new features, measured by the average time it takes from code commit to production release.
- ✔ Effective Monitoring and Alerting:
 - ✔ Utilizing CloudWatch and Splunk for application monitoring led to a 20% decrease in Mean Time To Resolution (MTTR) for critical incidents, demonstrating improved incident response times.
- ✔ Cost Optimization and Right Sizing:
 - ✔ Through the implementation of various cost optimization strategies, the application realized a 15% reduction in monthly infrastructure costs. This reduction was achieved by leveraging Reserved Instances, Spot Instances, and AWS Cost Explorer recommendations.
- ✔ High Availability and Resilience:
 - ✔ The multi-region architecture with active/warm standby clusters contributed to a 99.99% availability rate, measured by the percentage of time the application remained accessible to users for over a quarter.
- ✔ Scalability and Performance Improvements:
 - ✔ By leveraging ECS Auto Scaling and dynamic resource allocation, the application achieved a 50% increase in throughput during peak usage periods, ensuring optimal user experience.
- ✔ Efficient Data Management and Migration:
 - ✔ The adoption of AWS services for data management resulted in a 60% reduction in data migration time, measured by the time it takes to transfer on-premises data to AWS S3 and RDS.
- ✔ Streamlined Application Lifecycle:
 - ✔ The integration of AWS services for DevOps, monitoring, and deployment automation led to a 30% reduction in application release cycle time, measured by the average duration between consecutive production releases.